

PATENT ABSTRACTS OF JAPAN

(11) Publication number : 2000-010840

(43) Date of publication of application : 14. 01. 2000

(51) Int. Cl.

G06F 12/00

G06F 12/14

G06F 17/21

(21) Application number : 10-179214

(71) Applicant : NEC SOFTWARE CHUGOKU LTD

(22) Date of filing : 25. 06. 1998

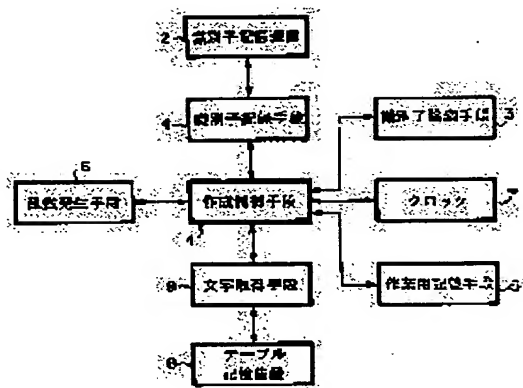
(72) Inventor : IWATA NORIHIKO

(54) IDENTIFIER GENERATION SYSTEM FOR COMPUTER SYSTEM

(57) Abstract:

PROBLEM TO BE SOLVED: To provide a function for generating a unique random identifier for a system which manages identifiers on a computer system.

SOLUTION: This system is provided with a clock 7 which imparts current time, a random number generating means 6 which generates a random number for identifier generation, a table storage device 8 which manages the relation between the date and time given by the clock 7 and the random number generated by the random number generating means, a character acquiring means 9 which acquires characters for an identifier from the table storage device 8, a working storage means 5 which stores an identifier in process of generation, an identifier recording means 4 which records the generated identifier in the identifier storage device, an identifier storage device 2 which manages generated identifiers, and a generation control means 1 which controls identifier generating processes for respective elements. And, generation date and time information having been converted by using the random number is embedded in the identifier.



LEGAL STATUS

[Date of request for examination] 25. 06. 1998

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 2908441

[Date of registration] 02. 04. 1999

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C) ; 1998, 2003 Japan Patent Office

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開2000-10840

(P2000-10840A)

(43)公開日 平成12年1月14日(2000.1.14)

(51)Int.Cl. ⁷	識別記号	F I	マークト [*] (参考)
G 0 6 F 12/00	5 2 0	G 0 6 F 12/00	5 2 0 E 5 B 0 0 9
12/14	3 2 0	12/14	3 2 0 A 5 B 0 1 7
17/21		15/20	5 7 0 R 5 B 0 8 2
			5 7 0 C

審査請求 有 請求項の数2 O L (全 11 頁)

(21)出願番号 特願平10-179214

(22)出願日 平成10年6月25日(1998.6.25)

(71)出願人 000211329

中国日本電気ソフトウェア株式会社

広島県広島市南区稲荷町4番1号

(72)発明者 岩田 徳彦

広島県広島市南区稲荷町4番1号 中国日

本電気ソフトウェア株式会社内

(74)代理人 100065385

弁理士 山下 穰平

Fターム(参考) 5B009 SA14

5B017 AA08 BA07 BB02 CA16

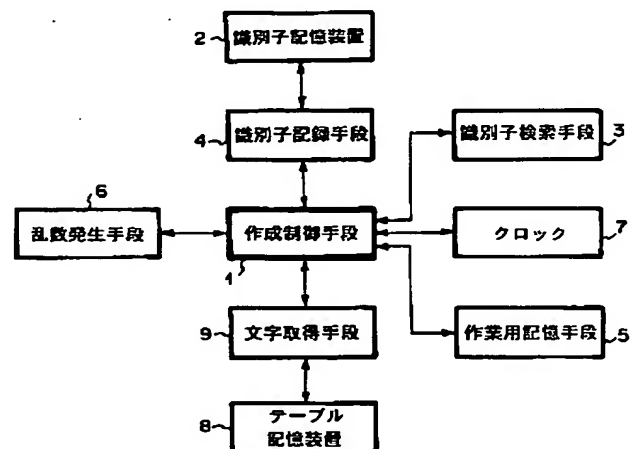
5B082 EA07 EA10 GA02 GA20

(54)【発明の名称】 コンピュータシステムにおける識別子発生システム

(57)【要約】

【課題】 識別子内に作成日時を作成日時を入れずに乱数によって識別子を作成すると、ある期間中に発行した識別子を全て管理しておき、二重発行のチェックを実施しなければならず、また簡単な文字置換のみ実施して識別子に入れた場合には、セキュリティ力が弱い。

【解決手段】 現在時刻を与えるクロックと、識別子作成用の乱数を発生する乱数発生手段と、クロックにより与えられた日時および乱数発生手段で発生した乱数の関連を管理するテーブル記憶装置と、テーブル記憶装置から識別子用の文字を取得する文字取得手段と、作成途中の識別子を記憶しておく作業用記憶手段と、作成した識別子を識別子記憶装置に記録するための識別子記録手段と、既作成の識別子を管理する識別子記憶装置と、各要素の識別子作成処理を制御する作成制御手段とを設け、識別子中に、乱数を用いた変換を施した作成日時情報を埋め込むようにした。



【特許請求の範囲】

【請求項 1】 現在時刻を与えるクロックと、識別子作成用の乱数を発生する乱数発生手段と、前記クロックにより与えられた日時および前記乱数発生手段で発生した乱数の関連を管理するテーブル記憶装置と、前記テーブル記憶装置から識別子用の文字を取得する文字取得手段と、作成途中の識別子を記憶しておく作業用記憶手段と、作成した識別子を前記識別子記憶装置に記録するための識別子記録手段と、既作成の識別子を管理する識別子記憶装置と、前記各要素の識別子作成処理を制御する作成制御手段と、を備え、識別子中に、乱数を用いた変換を施した作成日時情報を埋め込むようにしたことを特徴とするコンピュータシステムにおける識別子発生システム。

【請求項 2】 指定識別子を検索する識別子検索手段を備え、この指定識別子が前記識別子記憶手段にすでに記憶されているかどうかを検索し、すでに記憶されているときには前記の識別子発生処理を再実行するように構成されている請求項 1 に記載の識別子発生システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、コンピュータシステムにおいて、識別子内に作成日時情報を埋め込むことができる識別子発生システムに関する。

【0002】

【従来の技術】 コンピュータシステムにおいて、識別子の管理は重要であり、識別子を利用してファイルの管理を行う技術は従来から多数提案されている。たとえば特開平 9-223130 号公報には、文書と文書識別子とを互いに対応づけて格納する文書格納手段と、要求に応じて未発行の文書識別子の中の一つの文書識別子を発行するとともに、発行した文書識別子を含む管理情報を作成し、文書格納手段への文書の格納及び文書格納手段からの文書の取出に応じて、対応する文書識別子の管理情報のフラグを更新す識別子管理手段とを有し、この文書格納手段は、入力された文書が文書識別子を有するか否かを判定する手段と、入力文書が文書識別子を有しない場合には、舩認識別子管理手段に対し新規文書識別子の発行を要求し、この要求に応じて発行された文書識別子を入力文書とを対応づけて格納し、入力文書が文書識別子を有する場合には、識別子管理手段に対して当該文書識別子が文書格納手段に格納されているか否かを問い合わせ、この問い合わせの結果、当該文書識別子が格納されている場合に入力文書の格納を中止する手段と、格納している文書が取り出された場合に、識別子管理手段にその旨を通知する手段とを有するシステムが開示されている。

【0003】

【発明が解決しようとする課題】 上記のような従来の識別子作成処理では、識別子内に作成日時を入れることは

ないか、入れる場合も無変換もしくは簡単な文字置換のみを実施しているに過ぎない。したがって、この従来技術には、次のような問題点が生じていた。

【0004】 第 1 の問題点は、識別子内に作成日時を入れない場合、管理が容易ではないということである。その理由は、作成日時を入れずに乱数によって識別子を作成する場合、1 ヶ月以内は同一識別子を発行してはならない場面等においては、その期間中に発行した識別子を全て管理しておき、二重発行のチェックを実施しなければならないからである。

【0005】 第 2 の問題点は、作成日時を簡単な文字置換のみ実施して識別子に入れた場合、セキュリティ力が弱く、第三者に識別子中の作成日時部分の生成方法を知られてしまうということである。

【0006】 本発明は、以上の問題点を解決する識別子作成システムを提供することを目的としている。

【0007】

【課題を解決するための手段】 本発明は、コンピュータシステム上で識別子を管理するシステムにおいて、一意なランダム識別子を作成する機能を提供するものである。

【0008】 識別子に作成日時部分および乱数部分を有するようにすることで、同一時間以外では同じ識別子は作成されないようにする。またこれにより、識別子から作成日時を知ることが可能となる。

【0009】 前述の従来の技術で述べたように、作成日時を無加工のまま識別子内に入れた場合には、第三者にも識別子作成処理内容を知られる危険性が高くなり、機密性を必要とする場面では使用できなくなる。またこの方法では同一時間には、乱数部分の組み合わせ可能数分しか識別子を作成することが出来ない。

【0010】 一方、本発明では、作成日時を識別子内に入れる際にも乱数を利用しているので、作成方法の第三者への隠蔽、および同一時間の作成可能識別子数の増加が可能となる。

【0011】

【発明の実施の形態】 本発明の実施の形態の一例について図面を参照して説明する。

【0012】 図 1 に示す識別子発生システムは、各手段を利用して識別子作成処理を実施する作成制御手段 1 と、既作成の識別子を管理する識別子記憶装置 2 と、指定識別子を検索する識別子検索手段 3 と、作成した識別子を識別子記憶装置に記録するための識別子記録手段 4 と、作成途中の識別子を記憶しておく作業用記憶手段 5 と、識別子作成用の乱数を発生する乱数発生手段 6 と、現在時刻を与えるクロック 7 と、日時と乱数の関連を管理するテーブル記憶装置 8 と、テーブル記憶装置 8 から識別子用に 1 文字を取得する文字取得手段 9 とで構成されている。

【0013】 作成制御手段 1 は、各手段を利用すること

10

20

30

40

50

3

で識別子を作成する、メイン機能を遂行する部分である。

【0014】識別子記憶装置2は、既に発行した識別子を記憶する。

【0015】識別子検索手段3は、検索する識別子を与えられ、識別子記憶手段2が記憶している全ての識別子の中から、指定された識別子を探し出し、発見の成否を結果として与える。

【0016】識別子記録手段4は、新しく作成した識別子を、識別子記憶装置2上に追加記録する。

【0017】作業用記憶手段5は、識別子の作成開始から作成終了の間、作成途中の識別子を記憶する。

【0018】乱数発生手段6は、0～指定数の間の数をランダムに発生させる。

【0019】クロック7は、現在日時を返す。

【0020】テーブル記憶装置8は、日時と乱数から識別子用の文字を割り当てるための情報を2次元テーブルとして管理している。

【0021】文字取得手段9は、数値の組を与えられ、テーブルから対応する文字を1字取得する。

【0022】このように構成された本発明の第1の実施の形態における識別子発生システムの動作について、図2、図3を参照して詳細に説明する。

【0023】ここでは、識別子を「作成日時部分」7文字、「乱数部分」3文字の形10文字で形成する場合を例にとって説明する。「作成日時」は「YYYYMMDDhhmms」の形式で取得する（ただし「YYYY」部分は使用しない）。なお、ここで「%」は、剰余を求める演算を意味する。例えば、 $(8\%3=2)$ の形態をとる。

【0024】本発明の識別子作成システムにおいては、図2に示すように、最初に現在日時を取得し（ステップA1）、つぎに、識別子のうち、「作成日時」部分を作成する（ステップA2）。

【0025】つぎに図3において、乱数を1つ取得し（ステップB1）、現在日時のうち、「月」と「分」の値とともに、下記のテーブル2より識別子用の文字を一つ決定する（ステップB2）。行番号は $(月+(分\%12))\%12$ 、列番号は乱数 $\%3$ とする。取得した1文字を作業用領域（1文字目）に保存する。

【0026】

【表1】

10

20

30

40

4

テーブル2

	0	1	2
0	a	m	y
1	b	n	z
2	c	o	0
3	d	p	1
4	e	q	2
5	f	r	3
6	g	s	4
7	h	t	5
8	i	u	6
9	j	v	7
10	k	w	8
11	l	x	9

次に現在日時のうち、「日」と「秒」の値を使用し、下記のテーブル3より1文字を取得する（ステップB3）。行番号は $(日+(秒\%31))\%31$ とする。取得した1文字を作業用領域（2文字目）に保存する。

【0027】

【表2】

テーブル3

0	a
1	b
2	c
3	d
4	e
5	f
6	g
7	h
8	i
9	j
10	k
11	l
12	m
13	n
14	o
15	p
16	q
17	r
18	s
19	t
20	u
21	v

6

22	w
23	x
24	y
25	z
26	0
27	1
28	2
29	3
30	4

20 次に現在日時のうち、「時」と「秒」の値を使用し、テーブル3より1文字を取得する(ステップB4)。行番号は、 $(\text{時} + (\text{秒} \% 31)) \% 31$ とする。取得した1文字を作業用領域(3文字目)に保存する。

【0028】新しく乱数を取得する(ステップB5)。現在日時のうち「分」の10の位の値を行番号として使用し、下記のテーブル1より1文字取得する(ステップB6)。列番号は乱数%4とする。取得した1文字を作業用領域(4文字目)に保存する。

【0029】

30 【表3】

(5)

特開2000-10840

7
テーブル1

	0	1	2	3
0	a	k	u	4
1	b	l	v	5
2	c	m	w	6
3	d	n	x	7
4	e	o	y	8
5	f	p	z	9
6	g	q	0	0
7	h	r	1	1
8	i	s	2	2
9	j	t	3	3

8
テーブル4

0	a
1	b
2	c
3	d
4	e
5	f
6	g
7	h
8	i
9	j
10	k
11	l
12	m
13	n
14	o
15	p
16	q
17	r
18	s
19	t
20	u
21	v

新しく乱数を取得する（ステップB7）。現在日時のうち「分」の1の位の値を行番号として使用し、テーブル1より1文字取得する（ステップB8）。列番号は乱数%4とする。取得した1文字を作業用領域（5文字目）に保存する。

【0030】新しく乱数を取得する（ステップB9）。現在日時のうち「秒」の10の位の値を行番号として使用し、テーブル1より1文字取得する（ステップB10）。列番号は乱数%4とする。取得した1文字を作業用領域（6文字目）に保存する。

【0031】新しく乱数を取得する（ステップB11）。現在日時のうち「秒」の1の位の値を行番号として使用し、テーブル1より1文字取得する（ステップB12）。列番号は乱数%4とする。取得した1文字を作業用領域（7文字目）に保存する。

【0032】「乱数部分」については、3文字分についてそれぞれ乱数を発生させ、その36による剰余を求め、下記のテーブル4より1文字を取得する。取得した文字を作業用領域（8～10文字目）に保存する。

【0033】

【表4】

9	
22	w
23	x
24	y
25	z
26	0
27	1
28	2
29	3
30	4
31	5
32	6
33	7
34	8
35	9

以上の動作で、作業用領域に 10 文字の識別子が作成される。次に、この識別子が、識別子記憶装置に既登録でないかどうかを調べる。登録されていた場合、再び識別子作成処理を実施する。登録されていなかった場合、これを識別子記憶装置に登録し、処理を終了する。この作成法で作成した識別子は、作成日時情報を正確に所持している。

・識別子の 4 文字目とテーブル 1 から、分の 10 の位の値がわかる。

・識別子の 5 文字目とテーブル 1 から、分の 1 の位の値がわかる。

・識別子の 6 文字目とテーブル 1 から、秒の 10 の位の値がわかる。

・識別子の 7 文字目とテーブル 1 から、秒の 1 の位の値がわかる。

・識別子の 1 文字目と分とテーブル 2 から、月の値がわかる。

【0034】まず 1 文字目とテーブル 2 から、行番号がわかる。行番号を L ($0 \leq L \leq 11$) とすると、行番号の決め方より、 $L = (月 + (分 \% 12)) \% 12$ となる。これにより $月 + (分 \% 12) = 12n + L$ となる整数 n が存在する。いま、「分」の値

は判明しているので、 $(分 \% 12)$ の値は計算できる。これを k ($0 \leq k \leq 11$) とする。すると $月 = 12n + L - k$ 。月は 1 から 12 の値しかとらないので、この式を満たす n は一意に定まり、月の値がわかる。・識別子の 2 文字目と秒とテーブル 3 から、日の値がわかる。

【0035】まず 2 文字目とテーブル 3 から、行番号がわかる。行番号を L ($0 \leq L \leq 30$) とすると、行番号の決め方より、 $L = (日 + (秒 \% 31)) \% 31$ となる。これにより $日 + (秒 \% 31) = 31n + L$ となる整数 n が存在する。いま、「秒」の値は判明しているので、 $(秒 \% 31)$ の値は計算できる。これを k ($0 \leq k \leq 30$) とする。すると $日 = 31n + L - k$ 。日は 1 から 31 の値しかとらないので、この式を満たす n は一意に定まり、日の値がわかる。・識別子の 3 文字目と秒とテーブル 3 から、時の値がわかる。

【0036】まず 3 文字目とテーブル 3 から、行番号がわかる。行番号を L ($0 \leq L \leq 30$) とすると、行番号の決め方より、 $L = (時 + (秒 \% 31)) \% 31$ となる。これにより $時 + (秒 \% 31) = 31n + L$ となる整数 n が存在する。いま、「秒」の値は判明しているので、 $(秒 \% 31)$ の値は計算できる。これを k ($0 \leq k \leq 30$) とする。すると $時 = 31n + L - k$ 。時は 0 から 23 までの値しかとらないので、この式を満たす n は一意に定まり、時の値がわかる。上記方法で「作成日時 (MDDhhmmss)」を識別子から算出することが出来る。

【0037】次に、図 4 に示す本発明の第 2 の実施の形態について説明する。

【0038】この第 2 の実施の形態は、識別子の作成日時部分の作成に、「年」の 1 の位の値を使用している点で、第 1 の実施の形態と異なっている。

【0039】図 3 に示された第 1 の実施の形態では、識別子に作成日時中の月・日・時・分・秒の情報を埋め込むことにより、1 年の間に作成した識別子について、その作成日時を正確に管理していた。第 2 の実施の形態では、識別子の作成日時部分に「年」の下一桁部分を埋め込むことにより、10 年の間に作成した識別子について、その作成日時を管理可能とする。

【0040】第 2 の実施の形態の動作を図 4 を参照して詳細に説明する。

【0041】図 4 のステップ B1～B12 の動作は、図 3 に示された第 1 の実施の形態の各ステップ B1～B12 の動作と同一のため、説明は省略する。

【0042】ステップ B12 までの動作により、識別子中の 7 文字を作成する。

【0043】新しく乱数を取得する (ステップ B13)。現在日時のうち「年」の 1 の位の値を行番号として使用し、テーブル 1 より 1 文字取得する (ステップ B14)。列番号は乱数 $\% 4$ とする。取得した 1 文字を作業用領域 (8 文字目) に保存する。

【0044】図 2 における「乱数部分」作成の動作については、2 文字分についてそれぞれ乱数を発生させ、その 36 による剰余を求め、テーブル 4 より 1 文字取得す

る。取得した文字を作業用領域(9～10文字目)に保存する。

【0045】以上の動作で、作業用領域に10文字の識別子が作成される。次にこの識別子が、識別子記憶装置に既登録でないかどうかを調べる。登録されていた場合、再び識別子作成処理を実施する。登録されていなかった場合、これを識別子記憶装置に登録し、処理を終了する。

【0046】この作成法で作成した識別子は、作成日時情報を正確に所持している。

【0047】・月・日・時・分・秒の求め方は、図3における説明時と同一のため、説明は省略する。

【0048】・識別子の8文字目とテーブル1から、年の1の位の値がわかる。

【0049】上記方法で「作成日時(YMMDDhhmmss)」を識別子から算出することができる。

【0050】第2の実施の形態では、10年の間に作成した識別子について、作成日時情報を正確に管理できるという新たな効果を有する。

【0051】以上、「作成日時部分」に年の下一桁の情報を埋め込む場合について説明したが、同様の方法で、年の他の桁の情報を埋め込むこともできる。また「乱数部分」の桁数を増やし、同一時刻に作成可能な識別子種を増加させることもできる。この場合、識別子全体の文字数は増加する。また「作成日時部分」と「乱数部分」の文字順序および、テーブル1～テーブル4の文字順序・文字種については変更してもよい。

【0052】

【発明の効果】以上に説明したように、本発明によれば、識別子を見ればその作成日時がわかるため、識別子と別に作成日時を保存・管理する必要がなく、したがって識別子内に作成日時情報を埋め込むことができ、この結果、識別子の管理が容易になる。

【0053】また、作成日時の埋め込み時に、乱数を併用しているので、この結果、第3者には識別子から作成日時を取得することが困難になる。

【0054】さらに、乱数との対応テーブルを用いることで、同一時間に作成した識別子でも見た目を大きく変えることが可能となる。また作成日時部分さえ確立されていれば、その他の部分は全く独立して作成することが可能であるので、作成する識別子を作成日時部分と乱数部分に分離し、作成する識別子の文字数がある程度自由に変更することが出来る。

【図面の簡単な説明】

【図1】本発明の第1の実施の形態のコンピュータシステムにおける識別子発生システムの構成を示すブロック図。

【図2】図1の識別子発生システムの動作を示すフローチャート。

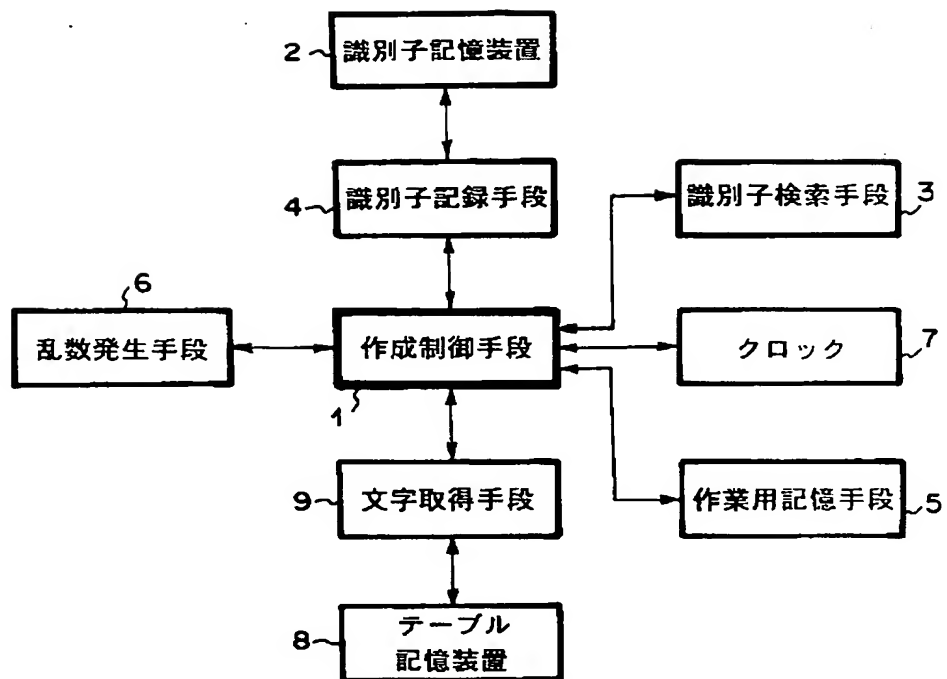
【図3】図1の識別子発生システムの動作を示すフローチャート。

【図4】本発明の第2の実施の形態のコンピュータシステムにおける識別子発生システムの動作を示すフローチャート。

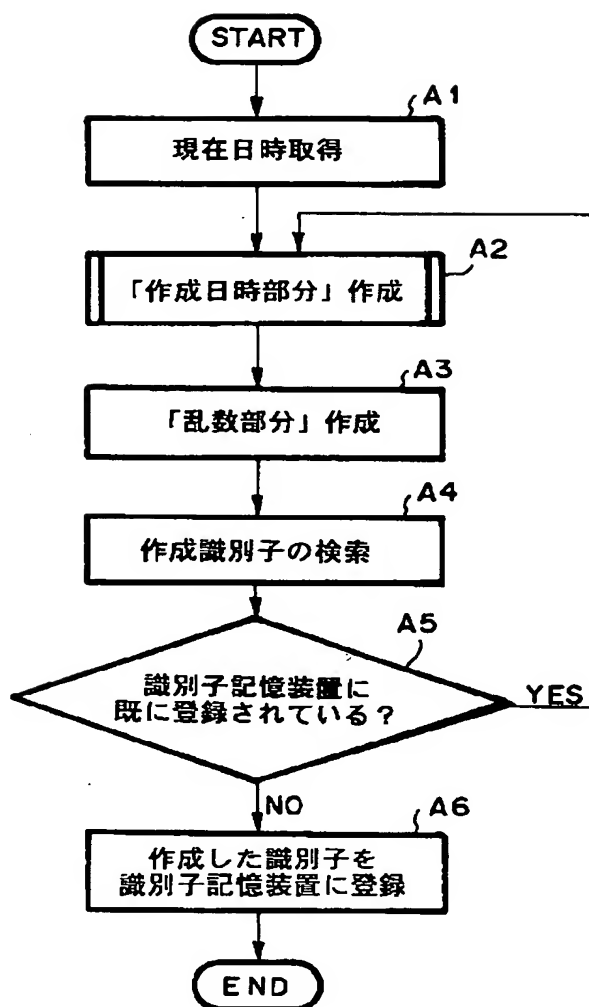
【符号の説明】

- 1 作成制御手段
- 2 識別子記憶装置
- 3 識別子検索手段
- 4 識別子記録手段
- 5 作業用記憶手段
- 6 乱数発生手段
- 7 クロック
- 8 テーブル記憶装置
- 9 文字取得手段

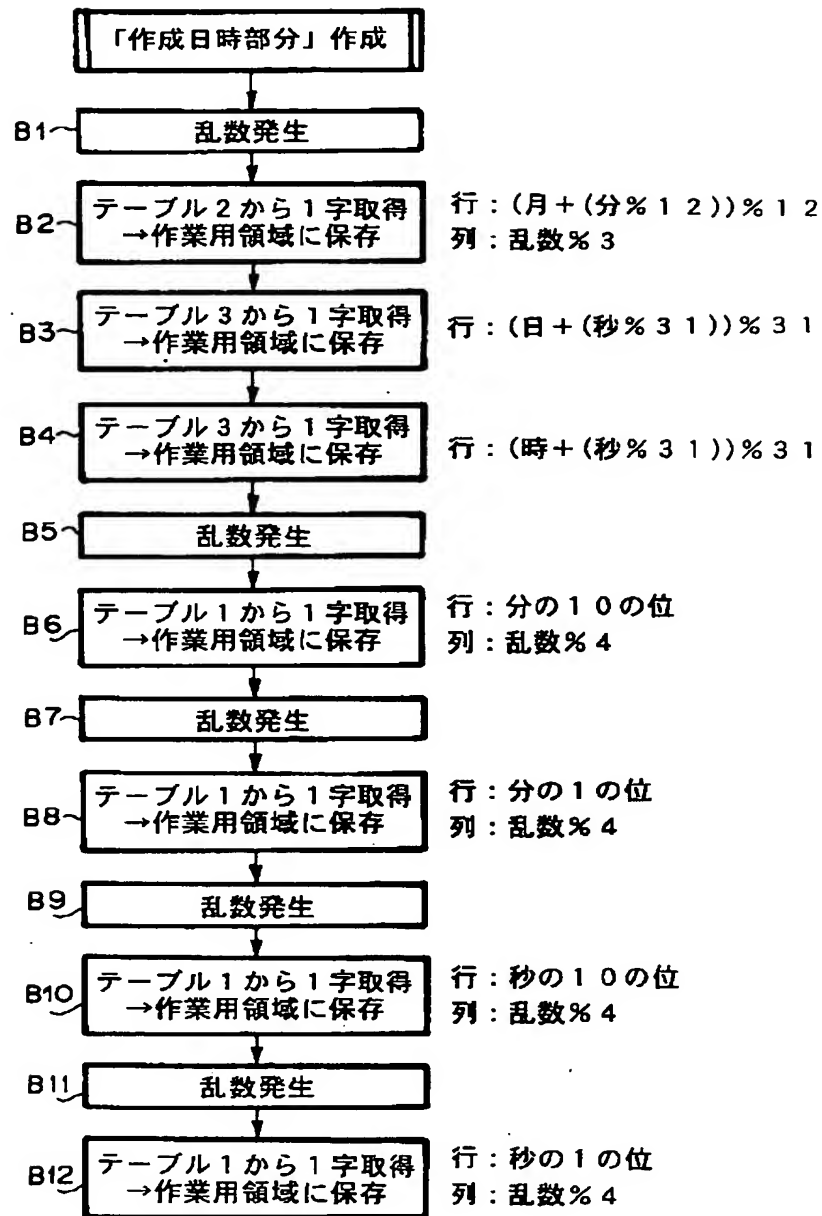
【図1】



【図 2】



【図 3】



【図 4】

